

Mark Ittensohn und Micha Rieser

Agile Produktentwicklung an Bibliotheken: Ein Erfahrungsbericht aus der Zentralbibliothek Zürich

Zusammenfassung: Der Bericht zeigt anhand einer Fallstudie aus der Zentralbibliothek Zürich auf, welche Vorteile und Herausforderungen die agile Produktentwicklung an Bibliotheken haben kann. Im Zentrum steht dabei das Rahmenwerk Scrum, welches zur Entwicklung des Fallbeispiels – eines Escape Games – genutzt wurde. Wie der Bericht nahelegt, hängt der Erfolg der agilen Produktentwicklung wesentlich vom Komplexitätsgrad des Produktes ab, sowie vom gemeinsamen Verständnis von Scrum und seiner Einbettung in andere Projektmanagement-Verfahren.

Schlüsselwörter:

Agile Product Development in Libraries: Report on a Case Study at the Zentralbibliothek Zürich

Abstract: On the basis of a case study at the Zentralbibliothek Zürich, the following report showcases the benefits and challenges that libraries may face when adopting agile product development. The primary focus lies on the framework Scrum, which was used for the development of the example in question: an escape game. The report suggests that the success of agile product development considerably depends on the complexity of a product, on a common understanding of Scrum as well as on its embedding within other methods of product development.

Keywords:

1 Einführung

Für viele Bibliotheken ist Agilität heutzutage kein Fremdwort mehr. Wie zahlreiche Publikationen der letzten Jahre belegen, sprechen sich immer mehr Stimmen für mehr Agilität im Bibliothekssektor aus. Gemeint ist damit oft nicht nur eine vage Bestrebung zu mehr Beweglichkeit, sondern die Adoption einer ganz konkreten Grundhaltung, die in der Privatwirtschaft schon seit mehreren Dekaden etabliert ist. Steve Dennings Definition von Agilität ist dafür repräsentativ:

Die Grundidee ist nicht einfach ein neuer Prozess oder eine neue Methode, sondern eine andere Ideologie – eine andere Art und Weise die Welt zu sehen und in der Welt zu handeln. Statt einer Ideologie von Kontrolle, mit einem Fokus auf Effizienz, Vorhersagbarkeit, detaillierten Plänen und internem Fokus, [ist Agilität] eine Ideologie der Befähigung, mit einem Fokus auf Selbstorganisation,

kontinuierlicher Verbesserung, iterativer Vorgehensweise, und, vor allem, dem ins Zentrum stellen des Kunden.¹

Erfolgsgeschichten, welche sich aus der Umsetzung dieser Art von Agilität im Bibliothekssektor ergeben, sind einige bekannt. So entwickelte die Bibliothek der Technischen Hochschule Chalmers in Schweden nicht nur ihre Webseite und ein institutionelles Repositorium auf agile Weise, sondern optimierte auch ihre Organisation durch die Adoption von agilen Werten und Prinzipien im Management.² Auch die Bibliotheken der finnischen Universität Tampere sowie die Bibliothek der Ohio State Universität führten in den letzten Jahren agile Ansätze im Management ein und steigerten damit ihre Fähigkeit, strategische Ziele zu erreichen.³ An der Bibliothek der Universität Colorado Boulder führten agile Prozesse zur Produktivitätssteigerung bei der Produktion von Digitalen Sammlungen,⁴ während die Bibliotheken der Universitäten Kansas und Minnesota durch agile Methoden grosse Erfolge in der Zusammenarbeit zwischen dem Bibliothekspersonal und den Forschenden erzielen konnten.⁵ Auch im deutschsprachigen Raum gibt es Beispiele. Die TIB in Hannover entwickelt zum Beispiel seit einigen Jahren erfolgreiche IT-Produkte mit agilen Werkzeugen,⁶ und in der Schweiz beschäftigen sich die Universitätsbibliothek Basel und die Zentralbibliothek (ZB) Zürich schon seit Jahren mit agilen Prozessen, Methoden und Denkmustern.

Die ZB Zürich ist dabei eine der wenigen Institutionen, welche physische Produkte agil entwickelt, d.h. nicht primär Software oder IT-Infrastruktur. Eines der ersten, und zugleich erfolgreichsten, dieser Produkte war das Escape Game «Schnebelhorners Vermächtnis». Das Produkt veranschaulicht, wie wichtig ein agiles Vorgehen bei der Entwicklung von physischen, bibliothekarischen Produkten sein kann, und welche Vorteile und Herausforderungen sich in der agilen Entwicklung ergeben können. Anhand von Einblicken in den Entwicklungsprozess des Escape Games liefert dieser Bericht einige Anhaltspunkte dazu, was es für Bibliotheken heissen kann, physische Produkte agil zu entwickeln. Dazu werden sowohl über die möglichen Beweggründe für agile Produktentwicklung reflektiert als auch Vorteile und Herausforderungen erläutert, die sich bei der praktischen Arbeit an physischen Produkten zeigen.

¹ Denning, 2015, keine Seitenzahlen, unsere Übersetzung.

² Forsman, 2014.

³ Niemi-Grundström, 2014; Jaggars und Jones, 2018.

⁴ Dulock und Long, 2015.

⁵ Lach und Rosenblum, 2018; McBurney et al., 2020; Wiggins et al., 2019.

⁶ Asch, keine Jahresangabe.

2 Agilität in Bibliotheken

Seit einigen Jahren findet Agilität auch in der Bibliothekswelt immer mehr Beachtung. Dies hat auch dazu geführt, dass Agilität teilweise als Hype oder Modewort verschrien wird.⁷ Es ist deshalb wichtig für eine adäquate Reflexion über agile Produktentwicklung in Bibliotheken, dass man sich bewusst wird, für welche Art von Problemen Agilität eigentlich Lösungen bieten kann. In der Privatwirtschaft wird die Bewegung zu mehr Agilität als Notwendigkeit angesehen, um mit der Wandelbarkeit der zeitgenössischen Welt umzugehen.⁸ Oftmals fällt in diesem Zusammenhang der Ausdruck «VUCA», ein Akronym der englischen Bezeichnungen «volatility» (Volatilität), «uncertainty» (Unsicherheit), «complexity» (Komplexität) und «ambiguity» (Mehrdeutigkeit).⁹ Die heutige Welt ist stark durch VUCA definiert. Sie steht damit im Kontrast zur zweiten Hälfte des 20. Jahrhunderts, als stabile gesellschaftliche und wirtschaftliche Rahmenbedingungen es einfach machten, langfristig zu planen sowie vorhandene Strukturen über lange Zeit bestehen zu lassen.¹⁰ In einer Umgebung, die durch steigende Dynamik und Komplexität charakterisiert ist, ist es jedoch besser für die wirtschaftliche Überlebensfähigkeit, rasch reagieren zu können und sich ständig den sich ändernden Umständen anzupassen. Die Adoption von Agilität, wie oben von Denning definiert, ist also kein Selbstzweck, sondern eine klare Reaktion auf die Charakteristika der heutigen Zeit.

Obwohl Bibliotheken meist keine privatwirtschaftlichen Unternehmen sind, müssen auch sie sich immer mehr mit dynamischen und zugleich disruptiven Veränderungen auseinandersetzen: sei es die fortschreitende Digitalisierung, die Globalisierung und die damit verbundenen demographischen Veränderungen, immer neue Informationstechnologien, der Wandel zu mehr kognitiver Arbeit und lebenslangem Lernen oder auch der Spardruck, welcher zunehmend auf Kultur- und Bildungsbudgets lastet.¹¹ Wie in der Privatwirtschaft zeigen sich auch in Bibliotheken traditionelle Ansätze, wie sie seit den 1970er zum Einsatz kommen, gegenüber dieser immerwährenden Veränderlichkeit als ineffektiv. Zu meinen gilt dies für die Ebene des Bibliotheksmanagement.¹² Zum anderen betrifft dies aber auch den deutlich konkreteren Bereich der Produktentwicklung. In Bibliotheken sind lineare Modelle oft noch vorherrschend, wie zum Beispiel sogenannte «Wasserfallmodelle». Diese sind darauf ausgelegt, zu Beginn die Anforderungen an ein Produkt abzuholen und dabei die für die Entwicklung nötigen Technologien und Kompetenzen zu bestimmen. In einem zweiten Schritt wird dann das Produkt

⁷ Oft vergessen wird dabei, dass die ersten Überlegungen zu Agilität in der Bibliothek schon über zwanzig Jahre alt sind: siehe Haricombe und Lusher, 1998, oder Tennant, 2001.

⁸ Siehe dazu, zum Beispiel, Goyk und Grote, 2018.

⁹ Siehe Bendel, 2019.

¹⁰ Siehe dazu zum Beispiel Grützner, 2019 S. 48.

¹¹ Siehe dazu zum Beispiel Grützner, 2019, S. 84; Mattmann, 2020, keine Seitenzahlen; Vonhof, 2018, S. 170-171.

¹² Vonhof, 2018, S. 174.

entwickelt. Wie Marco Nitschke betont, bedeutet dies, «dass am Anfang durch den Auftraggeber das Ergebnis beschrieben wird und er am Ende das bekommt, was der Auftragnehmer als Ergebnis verstanden hat».¹³ In einem solchen Entwicklungsprozess wird vor allem auf Planbarkeit gesetzt, und weniger beachtet, dass es in einer sich dynamisch und konstant verändernden Welt schwierig sein kann, die eigentlichen Anforderungen an Produkte, sowie die dafür nötigen Entwicklungstechnologien, vollständig im Voraus zu bestimmen. Um dieses Problem zu lösen, geht agile Produktentwicklung iterativ und inkrementell vor, anstatt linear einzelne Stadien und Gates zu durchlaufen: Nicht ein Gesamtentwurf und eine darauffolgende Entwicklung des Produkts findet statt, sondern ein Entwicklungsprozess in kurzen Zyklen, in welchen immer wieder nutzbare Ergebnisse produziert, evaluiert, und dann inkrementell aufeinander aufbauend weiterentwickelt werden.^[14] Daneben gehen agile Ansätze bei der Produktentwicklung auch davon aus, dass Entwickler*innen am besten wissen, wie sie das entwickeln können, was gewünscht ist. Daher ist neben der iterativen und inkrementellen Entwicklung selbstorganisiertes, teambasiertes Arbeiten ein wichtiger Teil der agilen Produktentwicklung, wie auch der Wille zur ständigen Verbesserung der Arbeitsorganisation.¹⁴^[15]

Wie bereits erwähnt, ist Komplexität eines der Charakteristika der modernen Welt. Für die agile Produktentwicklung ist dieser Punkt besonders wichtig. In der agilen Theorie wird Komplexität oftmals in einem präziseren Sinn verstanden, als dies allgemein üblich ist. In einer nach ihm benannten Matrix, kontrastiert der britische Management Professor Ralph D. Stacey vier mögliche Bereiche, in denen sich Produktentwicklung abspielen kann. Es sind dies einfach, kompliziert, komplex und chaotisch. Im Koordinatensystem der Stacey Matrix sind diese vier Bereiche halbkreisförmig entlang einer gedachten Winkelhalbierenden angeordnet:

¹³ Nitschke, 2019, S. 22

¹⁴ Pfeffer, 2020, S. 26.

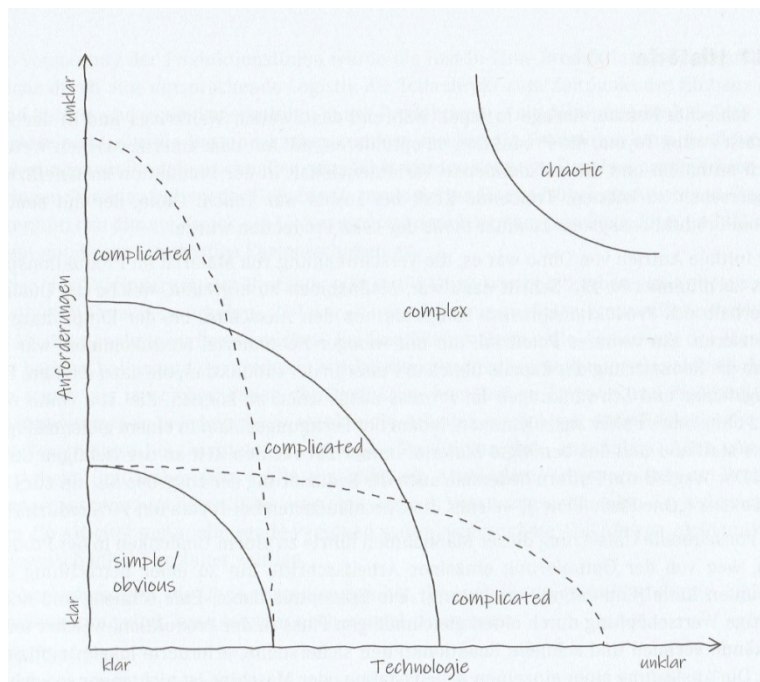


Abb. 1: Die Stacey-Matrix (Quelle: Pfeffer, 2020, S. 9.)

Laut Stacey sind in einfachen Situationen ('simple/obvious') sowohl die Anforderung als auch die Technologie an eine Produktentwicklung klar bekannt. Hier ist kein spezielles Vorgehen in der Entwicklung nötig. In komplizierten Situationen ('complicated') ist die Anforderung, die Technologie, oder auch beides, unklarer. Kompliziert bedeutet jedoch auch, dass Unklarheiten nicht so stark ausgeprägt sind, dass sie nicht durch Expertenwissen oder etablierte Praxis aufgeklärt werden können. Ein Beispiel für ein kompliziertes Produkt ist zum Beispiel ein Haus. Beim Hausbau bestehen explizite Regeln, welche zwar von Experten definiert sein müssen, jedoch von ihnen auch im Voraus geplant werden können.¹⁵ Im Gegensatz dazu sind in komplexen Umgebungen ('complex') die Anforderungen, die Technik, oder auch beides, so unklar, dass keine vorherrschende Praxis besteht und kein Expertenwissen alles abdecken kann. Ursachen und Wirkungen sind erst rückblickend ersichtlich. Aus diesem Grund ist es bei der Entwicklung von komplexen Produkten unabdingbar, immer wieder empirisch zu überprüfen, ob die Entwicklung auf einem richtigen Weg ist oder nicht, und sie basierend darauf kontinuierlich anzupassen. Nur so kann garantiert werden, dass man die Anforderungen stets berücksichtigt und Entwicklungstechnologien nutzt, die auch wirklich zielführend sind. Unter komplexen Voraussetzungen hat deshalb ein agiles Vorgehen in der Produktentwicklung klare Vorteile gegenüber einem linearen Entwicklungsprozess. Der Bereich des Chaotischen ('chaotic') beschreibt indes einen Zustand, in dem es unmöglich ist, überhaupt empirisch zu testen oder anzupassen. Dies trifft zum Beispiel dann zu, wenn es unkontrollierbar viele

¹⁵ Ripley und Miller, 2020, S. 139.

Stakeholder mit noch mehr verschiedenen Anforderungen gibt, oder keine Möglichkeiten oder Ideen für bestimmte Entwicklungstechnologien bestehen. Dies kann auch abhängig von den Kompetenzen eines Teams sein: zum Beispiel, wenn die Anforderungen nicht verstanden werden und die Technologie komplett unbekannt ist.



Abb. 2: Einblick in das Escape Game «Schnebelhorners Vermächtnis» (Quelle: Zentralbibliothek Zürich)

3 Das Escape Game der Zentralbibliothek Zürich

Als sich die Zentralbibliothek Zürich im Jahr 2019 entschied, ein Escape Game zu entwickeln, war es rasch ersichtlich, dass das Produkt im komplexen Bereich anzusiedeln war. Der Abteilung, die die Entwicklung übernahm – die IDE (Informationskompetenz, Digitale Dienste, Entwicklung) – fehlte jegliche Erfahrung in der Entwicklung solcher Spiele. Zwar wurden bereits an anderen Bibliotheken Escape Games entwickelt, jedoch stammen die meisten aus Bibliotheken mit unkomplizierten institutionellen Profilen sowie bekannter Kundschaft. Die Zentralbibliothek Zürich ist diesbezüglich komplexer aufgestellt. Als Stadt-, Kantons- und Universitätsbibliothek bedient sich nicht nur verschiedene Nutzergruppen (allgemeines, öffentliches Publikum bis hin zu Studierenden und Hobby-Forschenden), sondern besitzt auch ein Bestands- und Serviceprofile, das individuell auf diese verschiedenen Gruppen ausgerichtet ist. Auch das Escape Game sollte auf alle Angebotsprofile sowie Nutzergruppen der Zentralbibliothek passen. Diesbezüglich hätte qualifiziertes Expertenwissen, z.B. von Entwickler*innen anderer Escape Games an Bibliotheken, nur wenig Unterstützung geliefert, da

diese, wie erwähnt, oft sehr genau auf ein Bestandsprofil oder eine Nutzergruppe ausgerichtet sind. Dazu kam noch die Grundsatzfrage, ob ein Escape Game an der Zentralbibliothek Zürich überhaupt Anklang finden würde. Der öffentliche Markt wurde in den letzten Jahren vermehrt mit Spielen in diesem Format gesättigt. Zusätzlich war ein grosses Ziel des Produktes, eine weitere Zielgruppe zu gewinnen: unternehmungslustige Erwachsene mit Fachhochschulhintergrund. Für diese Zielgruppe wäre die ZB Zürich vom Bestand her interessant, aber bei den meisten Menschen dieser Gruppe nur peripher präsent. Das Escape Game schien vom Konzept her die ideale Lösung, diese Nutzergruppe in die Zentralbibliothek zu locken. Die Abteilung war sich bewusst, dass durch die vielen unbekannten Parameter es absolut essential war, möglichst früh, das heisst noch während der Entwicklungsphase, Benutzerfeedbacks einzuholen, sowie die eigene Entwicklungstechnologie konstant zu hinterfragen. Dazu fürchtete die IDE viele Risiken, die sie aus gegebenen Erfahrungsmangel noch nicht sehen und so nicht abschätzen konnte. Aus diesen Gründen zeigte sich ein agiles Vorgehen als wichtig, da es eine grösstmögliche Flexibilität in einer unsicheren Ausgangslage erlaubt, und die Chancen steigert, am Ende durch den agilen Prozess zu optimalen Ergebnissen zu kommen.

Was war das Ausgangskonzept, dass die ZB verfolgte? Ein Escape Game ist ein physisches Spiel, bei dem sich die Spieler in einer bestimmten Zeit durch das Lösen von verketteten Rätseln entweder aus einem geschlossenen Raum befreien müssen (Escape Room) oder etwas anderes aus einer geschlossenen Einheit befreien müssen. Bei dem Escape Game der ZB Zürich, «Schnebelhorner's Vermächtnis», müssen sich die Spieler nicht selbst befreien, sondern einen in einen Tresor eingesperrten Hund (in Form eines Stofftiers), der gemäss Vorgeschichte des Spiels sich aus Neugierde dort hinein verirrt hat und so versehentlich eingeschlossen wurde. Um die Rätsel innerhalb der gegebenen Zeit (60 Minuten) zu lösen, werden Kombinatorik, Geschicklichkeit und Informationsverhalten auf die Probe gestellt. Für das Thema des Escape Games wurde etwas gesucht, das die Zentralbibliothek Zürich repräsentieren kann und das in ihr Angebot passt. Da die ZB einerseits die SAC-Bibliothek seit 1890 im Bestand führt, ebenso aber auch viele Alpenpanoramen in Ihrer Kartensammlung hat, wurde das Alpenmotiv prägend.

Um den Turicensia-Sammelauftrag den Spielenden bekannt zu machen, erfanden das Entwicklerteam als prägenden Protagonisten einen Zürcher Alpinisten mit einem einzigartigen Namen Kunibert Helferich Schnebelhorner. (Anm. Das Schnebelhorn ist die höchste Erhebung im Kanton Zürich.) Die Vorgeschichte zum Escape Raum fasst sich so zusammen: Die Erben von Schnebelhorner vermachten seinen kompletten Nachlass der ZB unter der Bedingung, dass der Abtransport möglichst unverzüglich vonstattengeht. Während des Transports der Waren in ein Zwischenlager der Zentralbibliothek kam es allerdings zu einem Zwischenfall. Ein Hund der Zügelfirma schlich in einen

offenstehenden Tresor und wurde versehentlich eingeschlossen. Die Spieler*innen in Form eines Tierrettungsteam müssen nun möglichst schnell diesen Hund befreien, so dass ihm nicht der Erstickungstod droht. Um die Illusion aufzuhalten, wurde entschieden, dass dieser Fall nie eintreten sollte. Daher geben die Spielleiter*innen eingestreute Tipps, um zu garantieren, dass die Spielenden, falls die Rätsel für sie zu schwierig sind, die Lösung immer noch knapp vor Ablauf der Zeit erreichen.

4 Die Entwicklung des Escape Game

Wie genau ging die Bibliothek in der Entwicklung dieses Spiels agil vor? Agilität als Theorie kann in gewissen Belangen sehr vage klingen. In den letzten Jahren haben sich aber aufbauend von Definitionen von Agilität oder von agiler Produktentwicklung, konkrete Verfahren zur Umsetzung von Agilität etabliert. In der Produktentwicklung ist Scrum eines der bekanntesten Beispiele. Als Rahmenwerk ist es darauf ausgelegt, eine Grundstruktur zu liefern, welche es erlaubt, Grundzüge der Agilität in der Art und Weise wie man Produkte entwickelt praktisch umzusetzen. So hatte sich auch die Zentralbibliothek Zürich, genauer gesagt die für die Entwicklung des Escape Games zuständige Abteilung IDE entschieden, sich durch das Rahmenwerk Scrum für die Erarbeitung des Spieles leiten zu lassen.

Scrum postuliert, dass Entscheidungen immer auf der Grundlage von Beobachtungen getätigt werden sollen.¹⁶ Wie viele andere Verfahren der agilen Produktentwicklung setzt auch Scrum auf eine iterative und inkrementelle Entwicklung.¹⁷ Transparenz, Überprüfung und Anpassung sind dabei drei zentrale, empirische Säulen. Mit Transparenz ist gemeint, dass der Entwicklungsprozess sowie die entstehende Arbeit stets sichtbar ist für Entwickler und Auftraggeber. Transparenz ist die Grundlage für Überprüfung: der häufige und sorgfältige Vergleich zwischen Fortschritt und Zielen, sowie die Reflexion über mögliche Abweichungen und Probleme. Damit Überprüfung sinnvoll ist, verlangt Scrum, dass immer angepasst wird, zum Beispiel dort, wo «einzelne Aspekte eines Prozesses von akzeptablen Grenzen abweichen oder wenn das resultierende Produkt nicht akzeptabel ist».¹⁸

Aufbauend auf dieser theoretischen Grundlage umfasst Scrum ein Scrum Team (mit den Rollen Developer/Entwickler*innen, Product Owner und Scrum Master), fünf Arten von Events sowie drei verschiedene Artefakte. Das Scrum Team ist klein (üblicherweise zehn oder weniger Personen) und selbstorganisiert sowie interdisziplinär. Es ist umsetzungsverantwortlich für alle produktbezogenen Aktivitäten sowie ergebnisverantwortlich für die inkrementelle Produktion. Die Aufgabe der Rolle der

¹⁶ Schwaber und Sutherland, 2020, S. 3.

¹⁷ Schwaber und Sutherland, 2020, S. 3.

¹⁸ Schwaber und Sutherland, 2020, S. 4.

Entwicklerin oder des Entwicklers ist es, in jedem Sprint ein nutzbares Inkrement zu schaffen. Der Product Owner ist verantwortlich für die Maximierung des Produktnutzens, wie sich dieser aus der Arbeit des gesamten Scrum Teams ergibt.¹⁹ Als solcher legt er in Absprache mit Stakeholdern und Kunden das Produkt-Ziel fest und kommuniziert die Ziele des jeweiligen inkrementellen Entwicklungsschritts an das Entwicklungsteam. Der Scrum Master hilft dem Scrum-Team, die Scrum-Theorie und das Rahmenwerk zu verstehen und einzuhalten. Zusätzlich obliegt es dem Scrum Master, die Effektivität des Scrum-Teams zu steigern, sei es durch Coaching oder durch die Beseitigung von möglichen Hindernissen.

Individuelle Entwicklungsschritte finden bei Scrum in sogenannten Sprints statt, die bis zu vier Wochen dauern können, und die wiederum vier weitere Events einschliessen: Sprint Planning, Daily Scrum, Sprint Review und Sprint Retrospective. Im Sprint Planning werden die für den jeweiligen Sprint auszuführenden Arbeiten dargelegt und ausgewählt. Der Product Owner und die Entwickler*innen besprechen gemeinsam, warum ein jeweiliger Sprint wertvoll ist und welche Arbeiten abgeschlossen werden sollen. Der Scrum Master fungiert dabei sozusagen als Vermittler zwischen Entwicklungsteam und Product Owner. Dazu erstellt das Scrum Team ein Sprint Backlog aus einem vorhandenen Product Backlog. Während der Umsetzung und Realisierung des Sprint Backlogs trifft sich das Entwicklerteam jeden Tag zum Daily Scrum. In diesem 15-minütigen Treffen koordinieren die Entwickler*innen ihre Arbeit und nutzen den Austausch, um auf Hindernisse und Probleme zu reagieren. Am Ende des jeweiligen Sprints wird in der Sprint Review das Ergebnis überprüft und darauf aufbauend Anpassungen am Product Backlog gemacht. Dazu stellt das Scrum-Team das Inkrement den wichtigsten Stakeholdern vor und diskutiert gemeinsam die Fortschritte in Richtung des Produkt-Ziels. Schliesslich dient die Sprint Retrospective dazu, die Effektivität und Qualität des Entwicklungsprozesses sowie der Arbeitsorganisation zu steigern. Das Scrum-Team analysiert, wie der letzte Sprint verlaufen ist und schlägt mögliche Änderungen vor, welche die Effektivität des Teams verbessern würden.

Scrum-Artefakte repräsentieren «Arbeit oder Wert»²⁰ und bilden eine transparente Basis aufgrund welcher Überprüfung und Anpassung stattfinden kann. Der erste Artefakt ist das Product Backlog: eine Liste von Dingen, welche zur Erreichung des Produkt-Ziels nötig sind. Der Product Owner erstellt die Liste, priorisiert sie und stellt sicher, dass sie transparent ist. Das Product Backlog ist die zentrale Quelle der Arbeit, welche vom Entwicklerteam erledigt werden soll. Das zweite Artefakt, das Sprint Backlog, enthält die für den individuellen Sprint ausgewählten Product-Backlog-Einträge sowie einen

¹⁹ Schwaber und Sutherland, 2020, S. 6.

²⁰ Schwaber und Sutherland, 2020, S. 11.

Plan für deren Entwicklung. Schliesslich ist das Inkrement, das dritte Scrum-Artefakt, «ein konkreter Schritt in Richtung des Produkt-Ziels.»²¹ Ein oder mehrerer Inkremente werden jeweils in den Sprints geschaffen. Jedes Inkrement muss nach Abschluss stets verwendbar sein und zusammen – mit den bereits bestehenden – funktionieren. Arbeit kann nur als Teil eines Inkrements gesehen werden, wenn sie die Definition of Done erfüllt. Die Definition of Done umfasst die formale Beschreibung der Qualitätsstandards, welche jedes Inkrement erfüllen muss, um als abgeschlossen zu gelten.²² Als solches kann die Definition of Done entweder Qualitätsstandards der Organisation enthalten, oder es obliegt dem Scrum-Team, eine eigene «für das Produkt geeignete» Definition of Done zu erstellen.²³

Das Ziel von Scrum ist es nicht, detaillierte Anweisungen für alle möglichen Eventualitäten in der Produktentwicklung zu geben. Stattdessen definiert Scrum nur Elemente, welche «zur Umsetzung der Scrum-Theorie erforderlich sind».²⁴ Innerhalb von Scrum ist es möglich, andere Techniken oder Methoden einzusetzen und damit das Scrum-Rahmenwerk zu ergänzen. In der Praxis wird dies oft gemacht.²⁵ Auch Anwendungen von Scrum, welche nur Teile vom Rahmenwerk übernehmen oder Änderungen daran vornehmen, sind bei weitem keine Seltenheit. «Scrum But», wie solche Prozesse genannt werden, werden nicht nur oft adoptiert, sondern auch in der Fachliteratur sowie im Social Web rege diskutiert.²⁶ Dabei betonen Befürworter von «Scrum But» oft das Spannungsverhältnis zwischen Agilität und der dogmatischen Befolgung eines Regelwerks. «Scrum But»-Kritiker hingegen heben gerne Probleme hervor, welche sich aus einer Abweichung vom Scrum-Rahmenwerk ergeben können. Dies impliziert manchmal, dass die korrekte und regeltreue Anwendung von Scrum ohne Probleme auskommt, was natürlich ebenfalls nicht korrekt ist.

So zeigte sich auch in der Entwicklung des Escape Games, dass Scrum und Agilität eben nicht dasselbe sind, und dass sich auch bei vorherrschender Überzeugung, dass die Entwicklung eines solchen Produktes agil vorhergehen soll, durchaus auch Herausforderungen bei der Anwendung eines spezifischen Rahmenwerks zeigen können. Für die IDE war die Anwendung von Scrum bei der Entwicklung des Escape Games Neuland. Zwar versuchte das Team, im April und Mai 2019, unter Verwendung von Teilen von Scrum E-Learning Einheiten für das Romanische Seminar der Universität Zusammenstellen. Da allerdings die Entwicklung dieses Produkts nur gerade einen Sprint (d.h. drei Wochen) umfasste und die Anforderungen nicht besonders komplex waren, galt diese Anwendung

²¹ Schwaber und Sutherland, 2020, S. 13.

²² Schwaber und Sutherland, 2020, S. 13.

²³ Schwaber und Sutherland, 2020, S. 13.

²⁴ Schwaber und Sutherland, 2020, S. 3.

²⁵ Ein prominentes Beispiel dafür ist, zum Beispiel, «Scrumban».

²⁶ Preussig schätzt sogar, dass über 90% von Teams, die von sich sagen Scrum zu nutzen, in Wirklichkeit Scrum But praktizieren (2015, S. 154).

dem Kennenlernen und Aneignen eines zukünftigen agilen Entwicklungsprozesses. Die Abteilung sprach auch noch nicht von «Scrum», sondern bezeichneten das Vorgehen als «scrumish». Und zwar deshalb, weil die Aussage aus dem Scrum-Guide berücksichtigt werden sollte, dass ein Entwicklungsprozess erst dann «Scrum» genannt werden soll, wenn alle Teile des Rahmenwerks auch umgesetzt werden. So fehlte durch einen einzigen Sprint das iterative und inkrementelle Vorgehen und folgte somit einem Wasserfallmodell. Das nachfolgende Produkt des Escape Games bot sich aus den oben genannten Gründen aber an, das vollständige Rahmenwerk zu implementieren.

Das Escape Game «Schnebelhorners Vermächtnis» wurde in vier Sprints vom 21. Juni 2019 bis 27. November 2019 mit Scrum erstellt. Die nötigen Werkzeuge wurden ad hoc eingerichtet. So wurde das Fortschrittsdiagramm (Burn-Down-Chart) mit Excel berechnet und das Backlog wurde mit dem Onlineplanungswerkzeug Trello geführt. Die Ergebnisse aus den Tests wurden mit Excel festgehalten und ausgewertet.

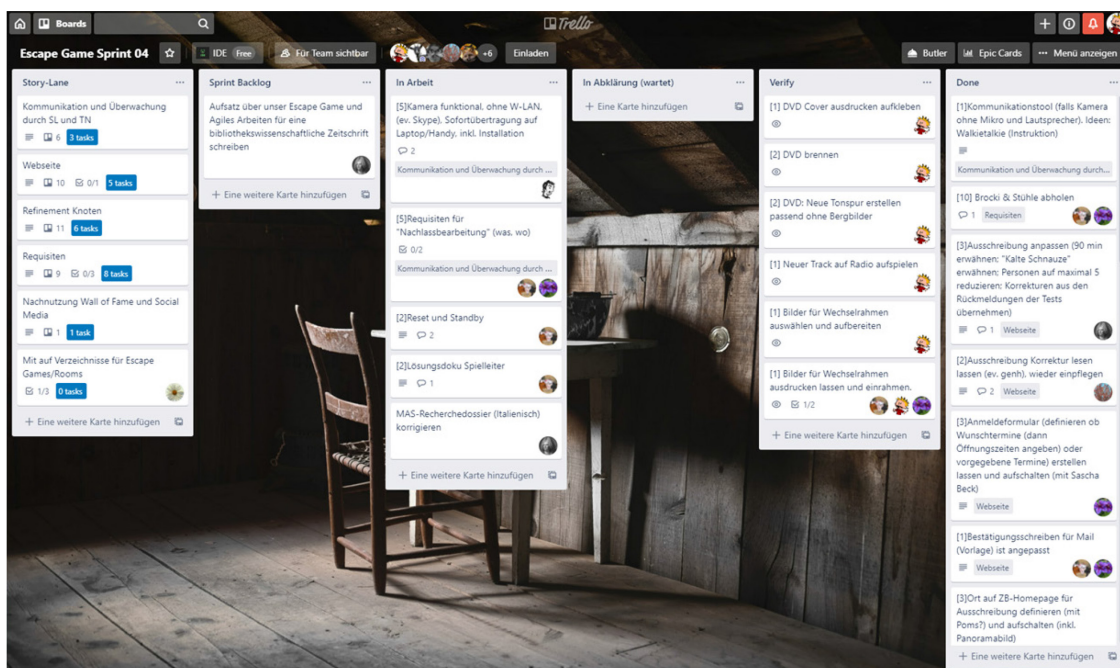


Abb. 2: Abwicklung von Tasks und Userstories mit Trello (Quelle: Zentralbibliothek Zürich)

5 Diskussion

Das war das erste Projekt der Zentralbibliothek, das Scrum vollständig umsetzen sollte. Retrospektiv betrachtet sah man im übergeordneten Projektvorgehen und in der Implementierung von Scrum einige Schwächen. In der folgenden Diskussion geht es daher nicht darum, ob die Mitarbeitenden in der Abteilung die einzelnen Scrum-Rollen korrekt ausgeführt haben, sondern ob die Erwartung der Abteilung gegenüber dem Werkzeug Scrum korrekt war. Man stellte fest, dass man viele der

Rahmenbedingungen in einer definierten Initialisierungsphase klären sollen. Beispielsweise welcher Raum für das Spiel genutzt werden sollte und welche Möglichkeiten er für die Umsetzung bietet, ob Budget zur Verfügung steht und wer letztendlich für die Kosten aufkommt, wo die handwerklichen Arbeiten vorstattengehen können und welche fachlichen Fähigkeiten überhaupt bei den Entwicklern vorausgesetzt werden müssen, damit auch die nichtbibliothekarischen Arbeiten sich gleichmässig aufteilen lassen. Es stellte sich heraus, dass eine hohe Anzahl von bestimmten Arbeiten nur von ganz wenigen spezialisierten Mitarbeitenden verrichtet werden konnte. Da es viele offene Punkte zu den Rahmenbedingungen gab, wurde die Klärung dieser als Aufgaben während der Entwicklung eingeplant. Damit wurden für die Entwicklungsphase fundamentale Risiken akzeptiert, die den Gesamterfolg hätten gefährden können. Die Erkenntnis wuchs, dass agiles Arbeiten nicht bedeutet, mit grösserer Unsicherheit starten zu dürfen, als eigentlich nötig wäre. Und dass nicht dank eines agilen Arbeitens die Initialisierungsphase eines Projektes schlicht ausgelassen werden darf. Es taucht während der Entwicklung ohnehin genug Unvorhergesehenes auf, auf das agil reagiert werden muss. Auch fehlte die teaminterne Auseinandersetzung über den Scope des Projektes. Es fehlte somit die Erarbeitung einer gemeinsamen Vorstellung, was denn das Escape Game alles bieten muss, welche Qualität es am Ende haben soll und was alles noch zum wesentlichen Kundennutzen gehört ('must') und was unwesentlich ist ('nice-to-have'). Der Leitstern war für das Team viel zu ambitioniert und zu wenig realistisch formuliert («bestes Escape Game im Raum Zürich»). Darum gab es während der Entwicklung einige Irritationen im Team, ob das Escape Game nun bereits kurz vor seiner Fertigstellung steht und nur noch der letzte Schliff getan werden muss oder ob das Projekt erst die Hälfte seiner Strecke zurückgelegt hat. Die Abteilung hat sich vermutlich deshalb zu Beginn zu wenig mit dem Scope auseinandergesetzt, da es zu sehr auf das in Scrum innewohnende inkrementelle Prinzip vertraut wurde. So glaubte das Scrum Team, dass nach jedem Sprint ein fertiges Produkt vorliegen würde und das Projekt so nach jedem Sprint als fertig erklärt werden könnte. Diese Vorstellung ist allerdings reichlich illusorisch. Da die Sprints nur kurz dauern (in der Regel drei Wochen), kann zu Beginn nicht so viel Arbeit erledigt worden sein, als dass bereits nach den ersten Iterationen ein fertiges Produkt in geforderter Qualität vorliegt. Dazu kommt, dass in den ersten Iterationen mehr Unsicherheiten zu klären sind, als in den folgenden, und die Entwickler*innen dadurch auch in den ersten Iterationen weniger produktiv sein können. Erst mit der Zeit wächst das Wissen, wie ein Escape Game erstellt werden muss und es entsteht die Sicherheit, dass auch das Richtige getan wird. Es ist deshalb auch für ein Scrum Team insgesamt von Vorteil, wenn es weiss, wieviel Zeit bis zum finalen Produkt eingeplant ist und nicht etwa die Erwartungshaltung kolportiert wird, als müsse das Team stets nach jeder Iteration ein fixfertiges Produkt abliefern können. Dass ein Inkrement stets ein nutzbares Produkt hervorbringt ist so zu definieren, dass dieses testbar ist und so

nutzbar wird und nicht, dass es bereits einen Endkundennutzen haben muss. So entspricht das Ergebnis der ersten Inkremente einem “Proof of Concept” und nicht einem fertigen Produkt. Diese Erkenntnis bewahrt den Project Owner vor der Illusion, als liefere Scrum fast zu beliebigen Zeitpunkten fertige Produkte und es brauche daher keine übergeordnete Projektplanung mehr. Eine unrealistische Erwartungshaltung von den Ergebnissen am Ende von Iterationen und auch das im Unklaren lassen von den übergeordneten zeitlichen Dimensionen führt zu Stress und zur Unruhe im Team.

Eines der Vorteile des agilen Verfahrens ist allerdings, dass das Team sehr früh Feedbacks aus Test mit internen und externen Personen bekam und es entweder frühzeitig etwas verbessern konnte oder dann Sicherheit gewann, das Richtige zu tun. So wurde sehr früh klar, dass die Rätsel von der Schwierigkeit her für die Zielgruppe lösbar waren, dass sie zeitlich nicht zu lang d.h. in einer Stunde lösbar und nicht zu kurz bemessen waren und dass sie den Teilnehmenden Spass machten. So verfestigte sich sehr bald der Eindruck, dass ein Escape Game entwickelt wurde, das sich in punkto Qualität mit kommerziellen Angeboten messen kann. Ein weiterer Vorteil ist, dass sich alle Entwickler trotz unterschiedlichen Fähigkeiten integriert fühlten und sich selbst organisieren konnten, da ein Arbeitsvorrat an Aufgaben bereitstand.

Die Abteilung IDE führte danach die agile Entwicklung fort und plante Sprints regelmässig ein. Auch wurde nun JIRA als professionelles Scrum-Werkzeug (im Gegensatz zu Trello und Excel) evaluiert und eingesetzt. So wurden bereits weitere Projekte wie Kundenforschungsprojekte oder Schulungsunterlagen für Studierende und Maturanden per Scrum entwickelt. Diese alle mit unterschiedlichem Erfolg, auch deshalb, weil bei einigen der Produktcharakter zweifelhaft war, oder sie kompliziert, aber nicht komplex waren. Da der Vorrat an Projekten stets kleiner wird und viele davon sich nicht ideal für Scrum eignen, ist nicht klar, wie nachhaltig Scrum in der Abteilung betrieben werden kann. Es ist zielführend, neue Produkte in einer Initialisierungsphase nach der Stacey-Methode zu analysieren und dann je nach Komplexitätsgrad sich für eine passende Umsetzungsvorgehen zu entscheiden, statt Scrum in einer Abteilung als Entwicklungsprozess vorauszusetzen und sämtliche Projekte auf diese Weise abzuwickeln.

Literaturverzeichnis

Asch, Malgorzata. Interview. <https://handschriftenportal.de/scrum-in-der-bibliothek> (keine Jahresangabe). Zugriff: 08.05.2021.

Bendel, Oliver. «VUCA.» Gabler Wirtschaftslexikon.

<https://wirtschaftslexikon.gabler.de/definition/vuca-119684/version-368877> (2019). Zugriff: 06.04.2021.

Denning, Steve. «More On Why Managers Hate Agile.»

<https://www.forbes.com/sites/stevedenning/2015/01/28/more-on-why-managers-hate-agile/?sh=12efb5bf10ea> (2015). Zugriff: 06.04.2021.

Dulock, Michael J. und Holley Long. «Digital Collections Are a Sprint, Not a Marathon: Adapting Scrum Project Management Techniques to Library Digital Initiatives.» *Information Technology and Libraries* (2015), S. 5-17.

Forsman, Daniel. "Introducing Agile Principles and Management to a Library Organization." *IATUL Annual Conference Proceedings*, Nr. 35 (2014), keine Seitenzahlen.

Goyk, R. und S. Grote. *Führungsinstrumente aus dem Silicon Valley*. Springer, 2018.

Grützner, Agnes. "Agile Führung – alter Wein in neuen Schläuchen oder Notwendigkeit für die Bibliothek der Zukunft?" *Perspektive Bibliothek*, Bd. 8 Nr. 1 (2019), S. 81-102.

Haricombe, Lorraine und T.J. Lusher. *Creating the Agile Library: A Management Guide for Librarians*. Greenwood Press, 1998.

Jaggars, Damon und DeEtta Jones. «An agile Planning and operations framework.» *Performance Measurement and Metrics*, Bd. 19, Nr. 2 (2018), S. 121-126.

Lach, Pamella und Brian Rosenblum. «Sprinting Toward Faculty Engagement: Adopting Project Management Approaches to Build Library-Faculty Relationships.» *Advances in Library Administration and Organization*, Bd. 38 (2018), S. 89-114.

Mattmann, Beat. «Was ist eigentlich diese Agilität, von der alle reden?» IGWBS Blog.

<https://www.igwbs.ch/was-ist-eigentlich-diese-agilitaet-von-der-alle-reden> (2020). Zugriff: 06.01.2021

McBurney, Jenny et al. «Library Research Sprints as a Tool to Engage Faculty and Promote Collaboration.» *Libraries and the Academy*, Bd. 20, Nr. 2 (2020), S. 305-338.

Niemi-Grundström, Minna. «Developing, evaluating and managing library with agile methods.» *Library Management*, Bd. 35, Nr. 6/7 (2014), S. 481-485.

Nitschke, Marco. *Agiles Projektmanagement in der Öffentlichen Verwaltung*. WvV, 2019.

Pfeffer, Joachim. *Produkt-Entwicklung: Lean & Agile*. Hanser, 2019.

Preussig, Jörg. *Agiles Projektmanagement: Scum, Use Case, Task Boards & Co*. Haufe, 2015.

Ripley, Ryan und Todd Miller. *Fixing Your Scrum: Practical Solutions to Common Scrum Problems*. The Pragmatic Bookshelf, 2020.

Schwaber, Ken, und Jeff Sutherland. «Der Scrum Guide.» <https://scrumguides.org/index.html> (2020).

Zugriff: 06.04.2021.

Tennant, Roy. «Building Agile Organizations.» *Library Journal Digital* (2001), keine Seitenzahlen.

Vonhof, Cornelia. «Bibliotheken und Agilität – Welten begegnen sich?» *Agile Verwaltung*, Hrsg. M Bartonitz et al. Springer, 2018. S. 169-183.

Wiggins, Benjamin et al. «Research Sprints: A New Model of Support.» *The Journal of Academic Librarianship*, Bd. 45 (2019), S. 420-422.

Dr. Mark Ittensohn
Zentralbibliothek Zürich – IDE
Zähringerplatz 6
CH-8001 Zürich
Switzerland
Orchid ID: <https://orcid.org/0000-0001-7239-0704>

Micha Rieser
Zentralbibliothek Zürich – IDE
Zähringerplatz 6
CH-8001 Zürich
Switzerland